Write a program to Implement Linear and Logistics regression

import numpy as np

import matplotlib.pyplot as plt

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression, LogisticRegression

# Generate synthetic data for Linear Regression

np.random.seed(42)

X\_linear = np.random.rand(100, 1) \* 10

y\_linear = 5 \* X\_linear + np.random.randn(100, 1) \* 3 # y = 5x + noise

# Generate synthetic data for Logistic Regression

X\_logistic = np.random.rand(100, 2) \* 10

y\_logistic = (X\_logistic[:, 0] + X\_logistic[:, 1] > 10).astype(int)

# Function to perform Linear Regression and plot results

def linear\_regression():

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_linear, y\_linear, test\_size=0.2, random\_state=42)

model = LinearRegression()

model.fit(X\_train, y\_train)

y\_pred = model.predict(X\_test)

# Plotting the regression line

plt.scatter(X\_test, y\_test, color='blue', label="Actual data")

plt.plot(X\_test, y\_pred, color='red', linewidth=2, label="Regression line")

plt.xlabel("X values")

plt.ylabel("Y values")

plt.title("Linear Regression")

plt.legend()

plt.show()

# Function to perform Logistic Regression and plot results

def logistic\_regression():

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_logistic, y\_logistic, test\_size=0.2, random\_state=42)

model = LogisticRegression()

model.fit(X\_train, y\_train)

# Plot decision boundary

x\_min, x\_max = X\_logistic[:, 0].min() - 1, X\_logistic[:, 0].max() + 1

y\_min, y\_max = X\_logistic[:, 1].min() - 1, X\_logistic[:, 1].max() + 1

xx, yy = np.meshgrid(np.linspace(x\_min, x\_max, 100), np.linspace(y\_min, y\_max, 100))

Z = model.predict(np.c\_[xx.ravel(), yy.ravel()])

Z = Z.reshape(xx.shape)

plt.contourf(xx, yy, Z, alpha=0.3)

plt.scatter(X\_test[:, 0], X\_test[:, 1], c=y\_test, edgecolors='k', marker='o', label="Test data")

plt.xlabel("Feature 1")

plt.ylabel("Feature 2")

plt.title("Logistic Regression Decision Boundary")

plt.legend()

plt.show()

# Run both regression functions

linear\_regression()

logistic\_regression()



